

IN THE CLAIMS:

Please substitute the following claims for the same-numbered claims in the application:

1. (Currently Amended) A method for incremental adaptation of a computer software application, said method comprising:

receiving a request for a component of said application from a device operated by a user of said application;

identifying, in response to said request, components of said application that may be requested by said user in the future, wherein the identifying process comprises graph analysis, statistical analysis, learning analysis, and response-time analysis, wherein said learning analysis comprises associating a penalty with an incorrect prediction of said components of said application that may be requested by said user in the future and biases a probability of selection of successive components more towards recently occurring historical patterns than older historical patterns, and wherein in said response-time analysis a maximum number of components in the neighborhood of a current component that can be adapted within a desired response-time are adapted by:

calculating the times required to adapt each respective component; and

given a maximum response time and starting from said current component, adding said times in a breadth first search order until a sum of the added times reaches said maximum response time without exceeding said maximum response time; and

adapting said identified components for operation with said requesting device,

wherein only said requested and identified components are adapted for operation with said

requesting device.

2. (Cancelled).

3. (Cancelled).

4. (Original) The method of claim 1, wherein components of said application comprise Internet webpages.

5. (Original) The method of claim 1, comprising the further step of:
performing, in response to said request, a reachability analysis to identify components reachable from said requested component; and
wherein said step of identifying comprises selecting components from said identified reachable components that are within a specified distance of said requested component.

6. (Original) The method of claim 5, wherein said specified distance comprises an integer value greater than or equal to one, said value representative of a number of transitions between two components of said application.

7. (Original) The method of claim 1, wherein said step of identifying comprises identifying components with a high probability of being requested based on historical request patterns relating to said application.

8. (Original) The method of claim 7, wherein said historical request patterns relate to requests from a substantially similar requesting device.

9. (Original) The method of claim 8, wherein said components are identified taking previously incorrect identification of components that may be requested into account.

10. (Previously Presented) The method of claim 1, wherein said step of identifying comprises identifying the maximum number of components neighboring said requested component that can be adapted within a specified period of time.

11. (Currently Amended) A method for incremental adaptation of a computer software application, said method comprising:

receiving a request for a component of said application from a device;

identifying, in response to said request, components reachable from said requested component, wherein the identifying process comprises graph analysis, statistical analysis, learning analysis, and response-time analysis, ~~and~~ wherein said learning analysis associates a penalty with an incorrect prediction of said components ~~of said application that may be requested by said user in the future~~ reachable from said requested component and biases a probability of selection of successive components more towards recently occurring historical patterns than older historical patterns, and wherein in said response-time analysis a maximum number of components in the neighborhood of a current component that can be adapted within a desired response-time are adapted by:

calculating the times required to adapt each respective component; and

given a maximum response time and starting from said current component, adding said times in a breadth first search order until a sum of the added times reaches said maximum response time without exceeding said maximum response time;

selecting components within a specified distance of said requested component from said identified reachable components; and

adapting said selected components for operation with said requesting device,

wherein only said requested and identified components are adapted for operation with said requesting device.

12. (Currently Amended) An apparatus for adaptation of a computer software application, said apparatus comprising:

at least one communications interface for transmitting and receiving data;

a memory unit for storing data and instructions to be performed by a processing unit; and

a processing unit coupled to said at least one communications interface and said memory unit, said processing unit programmed to:

receive a request for a component of said application from a device operated by a user of said application;

identify, in response to said request, components of said application that may be requested by said user in the future, wherein the identifying comprises graph analysis, statistical analysis, learning analysis, and response-time analysis, wherein in said response-time analysis a maximum number of components in the neighborhood of a current component that can be adapted within a desired response-time are adapted by:

calculating the times required to adapt each respective component; and

given a maximum response time and starting from said current component, adding said times in a breadth first search order until a sum of the added times reaches said maximum response time without exceeding said maximum response time;

associate a penalty with an incorrect prediction of said components of said application that may be requested by said user in the future and biases a probability of selection of successive components more towards recently occurring historical patterns than older historical patterns; and

adapt said identified components for operation with said requesting device,
wherein said processing unit is programmed to adapt only said requested and identified components for operation with said requesting device.

13. (Cancelled).

14. (Cancelled).

15. (Original) The apparatus of claim 12, wherein components of said application comprise Internet webpages.

16. (Original) The apparatus of claim 12, wherein said processing unit is further programmed to:

perform, in response to said request, a reachability analysis to identify components reachable from said requested component; and

identify reachable components for adaptation that are within a specified distance of said

requested component.

17. (Original) The apparatus of claim 16, wherein said specified distance comprises an integer value greater than or equal to one, said value representative of a number of transitions between two components of said application.

18. (Original) The apparatus of claim 12, wherein said processing unit is further programmed to identify components with a high probability of being requested based on historical request patterns relating to said application.

19. (Original) The apparatus of claim 18, wherein said historical request patterns relate to requests from a substantially similar requesting device.

20. (Original) The apparatus of claim 19, wherein said processing unit is further programmed to identify components taking previously incorrect identification of components that may be requested into account.

21. (Previously Presented) The apparatus of claim 12, wherein said processing unit is further programmed to identify the maximum number of components neighboring said requested component that can be adapted within a specified period of time.

22. (Currently Amended) An apparatus for adaptation of a computer software application, said apparatus comprising:

at least one communications interface for transmitting and receiving data;
a memory unit for storing data and instructions to be performed by a processing unit; and
a processing unit coupled to said at least one communications interface and said memory unit, said processing unit programmed to:

receive a request for a component of said application from a device;

identify, in response to said request, components reachable from said requested component, wherein the identifying comprises graph analysis, statistical analysis, learning analysis, and response-time analysis, ~~and~~ wherein said learning analysis associates a penalty with an incorrect prediction of said components ~~of said application that may be requested by said user in the future~~ reachable from said requested component and biases a probability of selection of successive components more towards recently occurring historical patterns than older historical patterns, and wherein in said response-time analysis a maximum number of components in the neighborhood of a current component that can be adapted within a desired response-time are adapted by:

calculating the times required to adapt each respective component; and

given a maximum response time and starting from said current component, adding said times in a breadth first search order until a sum of the added times reaches said maximum response time without exceeding said maximum response time;

select components within a specified distance of said requested component from said identified reachable components; and

adapt said selected components for operation with said requesting device,

wherein said processing unit is programmed to adapt only said requested and identified components for operation with said requesting device.

23. (Currently Amended) A computer program product comprising a computer readable medium comprising a computer program recorded therein for performing a method for incremental adaptation of a computer software application, said method comprising:

receiving a request for a component of said application from a device operated by a user of said application;

identifying, in response to said request, components of said application that may be requested by said user in the future, wherein the identifying process comprises graph analysis, statistical analysis, learning analysis, and response-time analysis, wherein said learning analysis comprises associating a penalty with an incorrect prediction of said components of said application that may be requested by said user in the future and biases a probability of selection of successive components more towards recently occurring historical patterns than older historical patterns, and wherein in said response-time analysis a maximum number of components in the neighborhood of a current component that can be adapted within a desired response-time are adapted by:

calculating the times required to adapt each respective component; and

given a maximum response time and starting from said current component, adding said times in a breadth first search order until a sum of the added times reaches said maximum response time without exceeding said maximum response time; and

adapting said identified components for operation with said requesting device,

wherein only said requested and identified components are adapted for operation with said requesting device.

24. (Cancelled).

25. (Cancelled).

26. (Original) The computer program product of claim 23, wherein components of said application comprise Internet webpages.

27. (Previously Presented) The computer program product of claim 23, wherein said method further comprises:

computer program code means for performing, in response to said request, a reachability analysis to identify components reachable from said requested component; and

computer program code means for identifying components for adaptation from said reachable components that are within a specified distance of said requested component.

28. (Original) The computer program product of claim 27, wherein said specified distance comprises an integer value greater than or equal to one, said value representative of a number of transitions between two components of said application.

29. (Previously Presented) The computer program product of claim 23, wherein said method further comprises identifying components with a relatively higher probability of being requested based on historical request patterns relating to said application.

30. (Original) The computer program product of claim 29, wherein said historical request

patterns relate to requests from a substantially similar requesting device.

31. (Original) The computer program product of claim 30, wherein said components are identified taking previously incorrect identification of components that may be requested into account.

32. (Previously Presented) The computer program product of claim 23 wherein said method further comprises identifying the maximum number of components neighboring said requested component that can be adapted within a specified period of time.

33. (Currently Amended) A computer program product comprising a computer readable medium comprising a computer program recorded therein for performing a method for incremental adaptation of a computer software application, said method comprising:

receiving a request for a component of said application from a device;

identifying, in response to said request, components reachable from said requested component, wherein the identifying process comprises graph analysis, statistical analysis, learning analysis, and response-time analysis, ~~and~~ wherein said learning analysis associates a penalty with an incorrect prediction of said components ~~of said application that may be requested by said user in the future~~ reachable from said requested component and biases a probability of selection of successive components more towards recently occurring historical patterns than older historical patterns, and wherein in said response-time analysis a maximum number of components in the neighborhood of a current component that can be adapted within a desired response-time are adapted by:

calculating the times required to adapt each respective component; and
given a maximum response time and starting from said current component, adding
said times in a breadth first search order until a sum of the added times reaches said maximum
response time without exceeding said maximum response time;
selecting components within a specified distance of said requested component from said
identified reachable components; and
adapting said selected components for operation with said requesting device,
wherein only said requested and identified components are adapted for operation with
said requesting device.